EXHIBIT QQQ

Part 3

$vSIM_{CORE}$ and $vSIM_{MGMT}$. Next, in phase 2, we consider subscriber authentication in GSM networks using the vSIM credential $Cred_{vSIM}$.

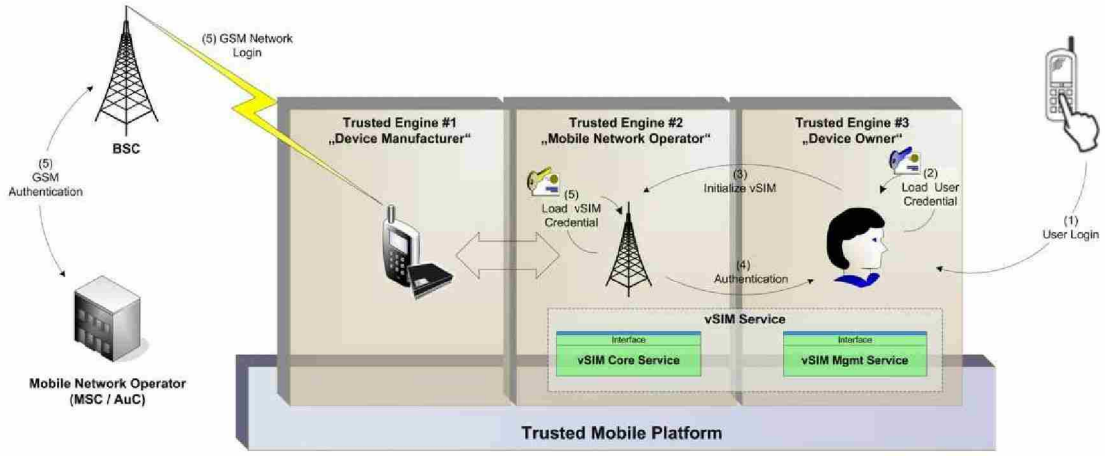

Figure 3.8: *Subscriber Authentication Figure - Model "One"*

Phase 1: "Initialization of vSIM Credentials" First, the user initialize the vSIM services and performs a log-in sequence. He/she sends a unique id ID_U with a proper password CHV_U to the $vSIM_{MGMT}$ service (Step 1), which loads the associated user credential from protected storage (Step 2).

$$U \rightarrow vSIM_{MGMT} \quad : \quad ID_U, CHV_U$$

Afterward, the $vSIM_{MGMT}$ service connects to the trusted interface layer of the $vSIM_{CORE}$ service and sends a vSIM credential initialization request to the $vSIM_{CORE}$ service (Step 3). After having received this request message, $vSIM_{CORE}$ generates a number $RAND_{AUTH}$, randomly chosen from a suitable range and sends this value as an authentication challenge to $vSIM_{MGMT}$, which holds the actual signature keys of U.

$$vSIM_{CORE} \rightarrow vSIM_{MGMT} \quad : \quad RAND_{AUTH}$$

Now, the $vSIM_{MGMT}$ takes the corresponding private portion of the user signature key, signs the challenge $RAND_U$ and sends this value back to the $vSIM_{CORE}$ service (Step 4).

$$vSIM_{MGMT} \rightarrow vSIM_{CORE} \quad : \quad SIGN_U(RAND_{AUTH})$$

Once, the $vSIM_{CORE}$ has received the signed message, it verifies its status. Finally, the $vSIM_{CORE}$ unseals $Cred_{vSIM}$ and initializes the SIM functionality using the $IMSI_i$ and K_i (Step 5).

Phase 2: "Subscriber Authentication" The GSM standard defines its own authentication protocol based on SIM credentials as described in Section 2.1.2. Since the SIM_{CORE} indirectly talks to the MNO, TSS_{DM} must provide a means to relay these messages between the $vSIM_{CORE}$ service and the MNO, this communication should be transparent to this protocol. All relevant communication mechanisms, like cryptographic algorithms A3 and A5, responsible for user authentication and key generation are implemented within the $vSIM_{CORE}$ module. [2]

Following protocol sequence outline the authentication process in GSM networks (Step 5, Step 6): First, the trusted platform initializes the authentication process and sends the $GSMAuthAlgorithm$ command to the $vSIM_{CORE}$ service of TE_{MNO}.

In the next step, the mobile device requests for authentication at the GSM network. Therefor, TSS_{DM} relays $IMSI_i$ (or $TMSI_i$) from $vSIM_{CORE}$ to MNO.

$$vSIM_{CORE} \rightarrow MNO \quad : \quad IMSI_i$$

The MNO generates internally a set of authentication triplets, as described in Section 2.1.2.1. It contains a authentication challenge $RAND_i$, a corresponding session key K_c and a $SRES$. The K_c and the $SRES$ are calculated with the GSM A38 algorithm. The MNO replies to TE_{MNO} by sending the challenge $RAND_i$.

$$MNO \rightarrow vSIM_{CORE} \quad : \quad RAND_i$$

This $RAND_i$ is passed to the trusted $vSIM_{CORE}$ service. Next, it also uses the A3 algorithm together with the key K_i. The output of the algorithm is the challenge response message $SRES^*$. The $vSIM_{CORE}$ sends this $SRES^*$ message to the MNO.

$$vSIM_{CORE} \rightarrow MNO \quad : \quad SRES^*$$

Finally, the MNO compares the $SRES$ with $SRES^*$. If they are equal, the subscriber is authenticated and $vSIM_{CORE}$ also derives the shared session key K_c

[2] Note that the specified GSM algorithms in phase 2 is substitutable by any other authentication algorithm, which requires provisioning of symmetric keys (e.g. one-time-password hashes or symmetric cryptographic keys) and associated attributes in form of a subscriber credential.

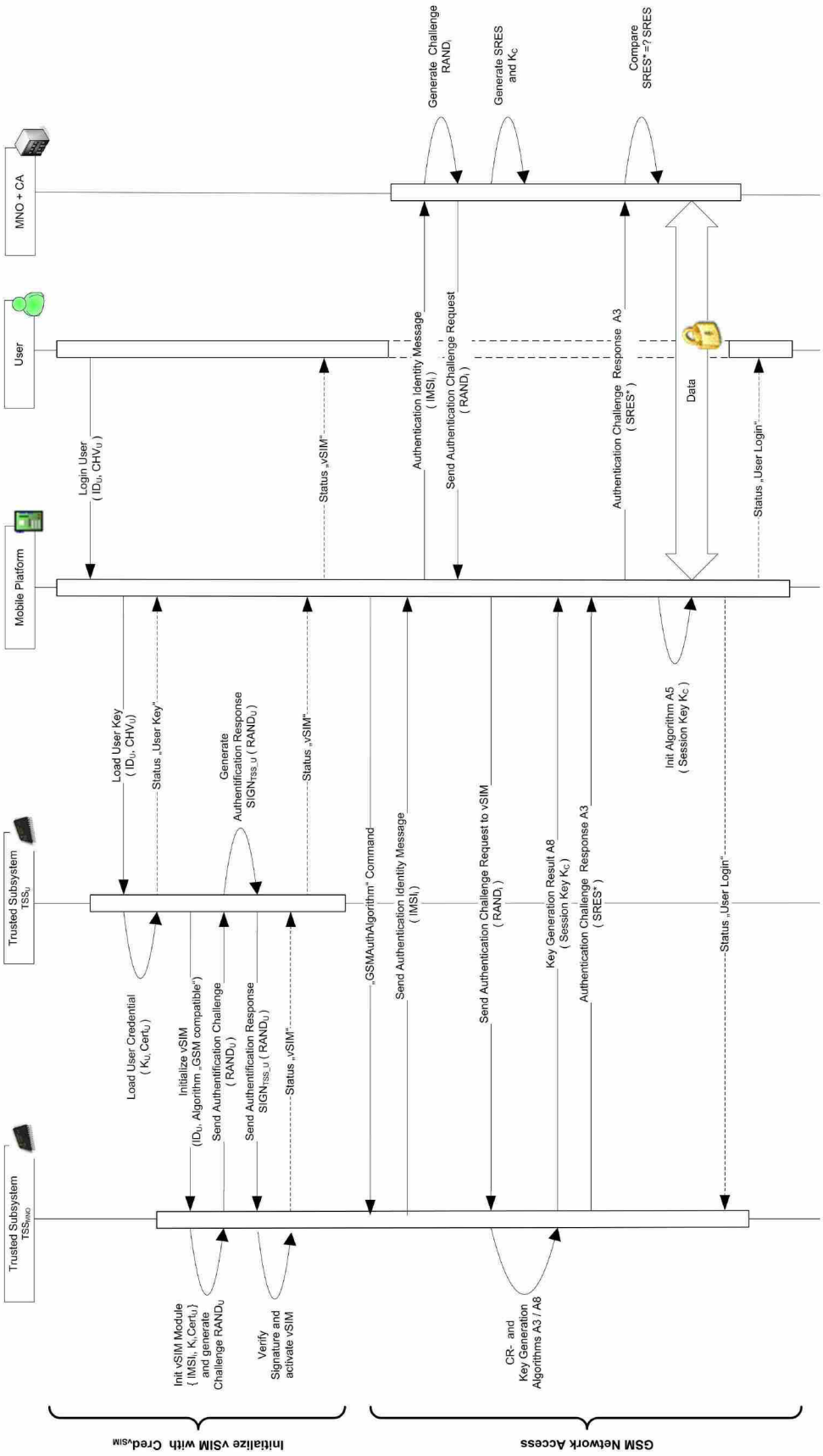

Figure 3.9: *Subscriber Authentication Protocol - Model "One"*

3.4.2 Model "Two" - Subscriber Authentication with Remote Attestation for Basic Network Access

In this section, we present a more comprehensive model compared with the precedent one. Additionally to model "One", we integrate remote attestation for basic network access. A variant of this method has been described in [KS06]. Beside the main task of SIM substitution, it provides

- user-authenticated access to the subscriber subdomain,
- device-authenticated access to a generic domain,
- mutual authentication between the MNO and a trusted mobile device.
- finer-grained functional restriction (e.g. SIM-lock), and
- dynamic down-/upgrade of services

As illustrated in Figure 3.10, all devices inside a generic domain are able to use the generic services of the mobile communication network. A trusted platform which is located in the MNO domain has access to both specific subscriber-authenticated services and generic services. Such generic service, for instance are location-based information or WLAN-based internet services. In case of a mobile phone is located

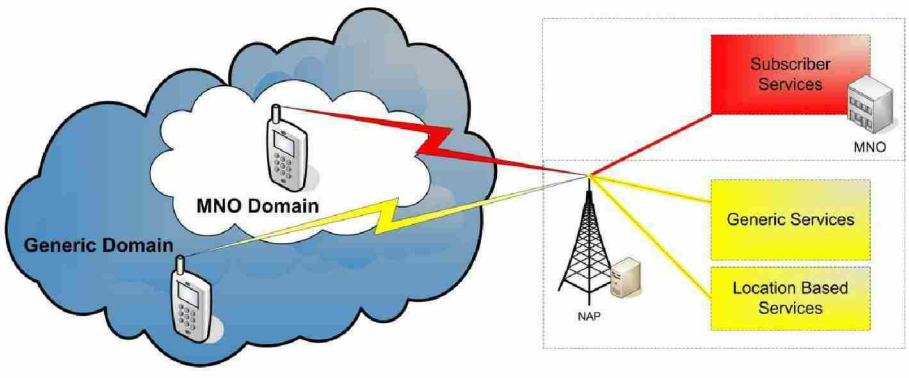

Figure 3.10: *Restricted Subdomain by Trust Credentials*

inside the generic domain, it uses a generic credential $Cred_{BASE}$ based on remote attestation mechanisms, to gain basic network access. The assignment to the subscriber domain of MNO is then done by performing a user-specific authentication process using vSIM credentials.

In model "Two", we offer two different ways for subscriber authentication. In phase 3, a similar approach to model "One" is described. Alternatively, phase 3"

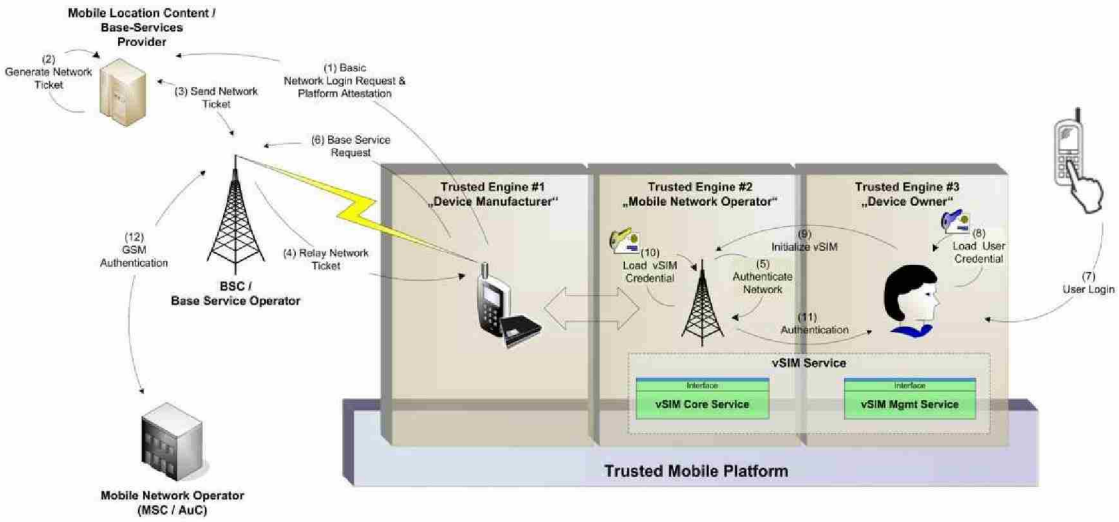

Figure 3.11: *Subscriber Authentication Figure - Model "Two"*

introduces another approach, which is build upon an established trust relationship of the generic domain.

3.4.2.1 Platform and Protocol Precondition

Similar to section 3.3.2, MTP has carried out the boot process and has loaded the specific OS and its trusted services. In particular, this also includes the vSIM services $vSIM_{CORE}$ and $vSIM_{MGMT}$. The trusted platform has checked, that the installed hardware and running software, are in a trustworthy state and configuration. It is able to report and attest this state, if challenged by an authorized entity.

3.4.2.2 Protocol Scheme and informal Description

This protocol description is separated into three phases as illustrated in Figure 3.11. The first phase describes the protocol for basic network access using remote attestation and ticketing, which is inspired by the Kerberos protocol [NYHR05]. In the second phase, the vSIM credential is initialized. Finally, the third phase holds the process for subscriber authentication.

Phase 1: "Basic Network Access" First, the trusted platform initialize the remote attestation and device authentication process. MTP requests the trusted engine TE_{DM} for a platform attestation and device authentication, addressed to the MNO. Then, the trusted engine TE_{DM} performs this request and connects to

the corresponding network access point NAP_{MNO} (Step 1). Therefor, the TSS_{DM} generates a random value $RAND_{BASE}$ and and performs a platform attestation. Next, the base authentication service of TE_{DM} sends $RAND_{BASE}$, the attestation data and its certificate $Cert_{DM}$ to the network access point.

$$\begin{aligned} TE_{DM} \rightarrow NAP_{MNO} \quad : \quad & RAND_{BASE}, Cert_{TSS_{DM}} \\ & ATTEST(S_i) \end{aligned}$$

Having received this request, the NAP_{MNO} checks the state of the client machine. If the signed integrity metric of the client platform fails verification or no reference state is found, the NAP_{MNO} aborts the protocol and replies with an error message. Otherwise, the platform passed authentication and is considered as trustworthy.

Afterward, the NAP_{MNO} requests an accredited entity to generates a session key K_{BASE} and a network ticket (Step 2). Such an accredited entity may be an authentication center AUC_{MNO}, which belongs to the mobile network provider MNO. Substantially, the ticket contains the following information:

$$\begin{aligned} Ticket_{BASE} \quad := \quad & \{ID_{MTP}, ID_{NAP}, K_{BASE}, \\ & REALM_{BASE}, LIFETIME_{BASE}\}. \end{aligned}$$

Next, AUC_{MNO} encrypts $Ticket_{BASE}$ with the public (or shared) encryption key K_{NAP} and send both, $Ticket_{BASE}$ and K_{BASE} to the NAP_{MNO} (Step 3), which relays it to the client platform (Step 4). Therefor, the message is bound to the trusted subsystem TSS_{DM} with the corresponding public key $K_{TSS_{DM}}$ and a valid platform state.

$$\begin{aligned} AUC_{MNO} \rightarrow TE_{DM} \quad : \quad & BIND_{K_{TSS_{DM}}}(K_{BASE}), \\ & ENC_{K_{NAP}}(Ticket_{BASE}), \\ & SIGN_{AUC_{MNO}}(RAND_{BASE}) \end{aligned}$$

Once, TSS_{DM} has received the signed message, it verifies the status of the signed $RAND_{BASE}$ (Step 5). If revoked, the subsystem replies with an error message and halts the protocol. Otherwise the AUC_{MNO} is authenticated by the challenge response.

Next, TSS_{DM} decrypts the session key K_{BASE} and sends $ENC_{K_{NAP}}(Ticket_{BASE})$ together with an authenticator A_{MTP} to the NAP_{MNO}. The authenticator A_{MTP}

is composed of its platform identity ID_{MTP}, the current network address $ADDR$, and a timestamp $TIME$.

$$TSS_{DM} \rightarrow NAP_{MNO} \quad : \quad ENC_{K_{NAP}}(Ticket_{BASE}), A_{MTP}$$

After, NAP_{MNO} has received the encrypted ticket, it verifies the embedded information. If the status is valid, the trusted platform is authenticated and access to the generic services is granted.

Phase 2:"Initialization of vSIM Credentials" The initialization of a vSIM credential is performed in Steps 7 - 11 of Figure 3.11. This process is identical to model "One". For a detailed description of the protocol sequence, we refer to Section 3.4.1.2.

Phase 3: "Subscriber Authentication" (Variant 1) Similar to Section 3.4.1.2, this variant performs subscriber access with compatibility to regular GSM authentication. In an additional step, K_{BASE} is substituted by the session key K_c on both sides, the NAP_{MNO} and MTP (Step 12).

However, this approach is optimizable, by embedding the $RAND_i$ already into the encrypted key message from Step 4. In this case, $vSIM_{CORE}$ extracts the $RAND_i$ from this message, calculates the challenge response $SRES$ and sends both to the MNO. The MNO generates internally the expected $SRES$ and the corresponding session key K_c.

At this point a mutual authentication between the AUC_{MNO} and U has been performed. The AUC_{MNO} is authenticated by the signed challenge, obtained in step 3.1. On the other hand, the user has prooven its identity by $SRES$. The authentication between NAP and U is implicitly proven by a valid communication key K_c.

If an explicit authentication of these entities is required, some additional steps have to be carried out. The NAP authenticates itself to the platform by the following steps. First the NAP extracts the timestamp from the authenticator A_U. Next, NAP increments the value and encrypts it with the shared communication key K_c (or a derivation of it). Finally, it sends the message back to the trusted platform.

Phase 3": "Subscriber Authentication" (Variant 2) Alternatively to phase 3, the following protocol sequence describes the authentication process in variation to standard GSM authentication.

Here, we envisage a slightly modified authentication method, which offers significant security enhancements across the entire PLMN. In particular protocol flaws in *Signaling System 7 (SS7)* could be bypassed.

It takes advantage of the former negotiated information from the device authentication in phase 1. In conventional GSM Infrastructures an authentication triplet is sent over the SS7 network. This triplet contains of a challenge $RAND$, the correct response $SRES$, and the communication key K_c.

While initial access to the mobile cellular network with the communication key K_{BASE} is still established, a renewal of this key is discretionary. In particular, embedding a communication key K_c within this token is not necessary. However, a specific realm and accordingly other specific service information has to be sent to the network access point NAP_{MNO}.

We note that this approach avoids transmission of unprotected communication keys K_c across the PLMN infrastructure. The main idea behind this model is to use the still established communication channel between NAP_{MNO} and MTP, which is protected by K_{BASE}. Instead of performing a renewal of the communication key, the MNO only sends a service update message to the respective network access point NAP.

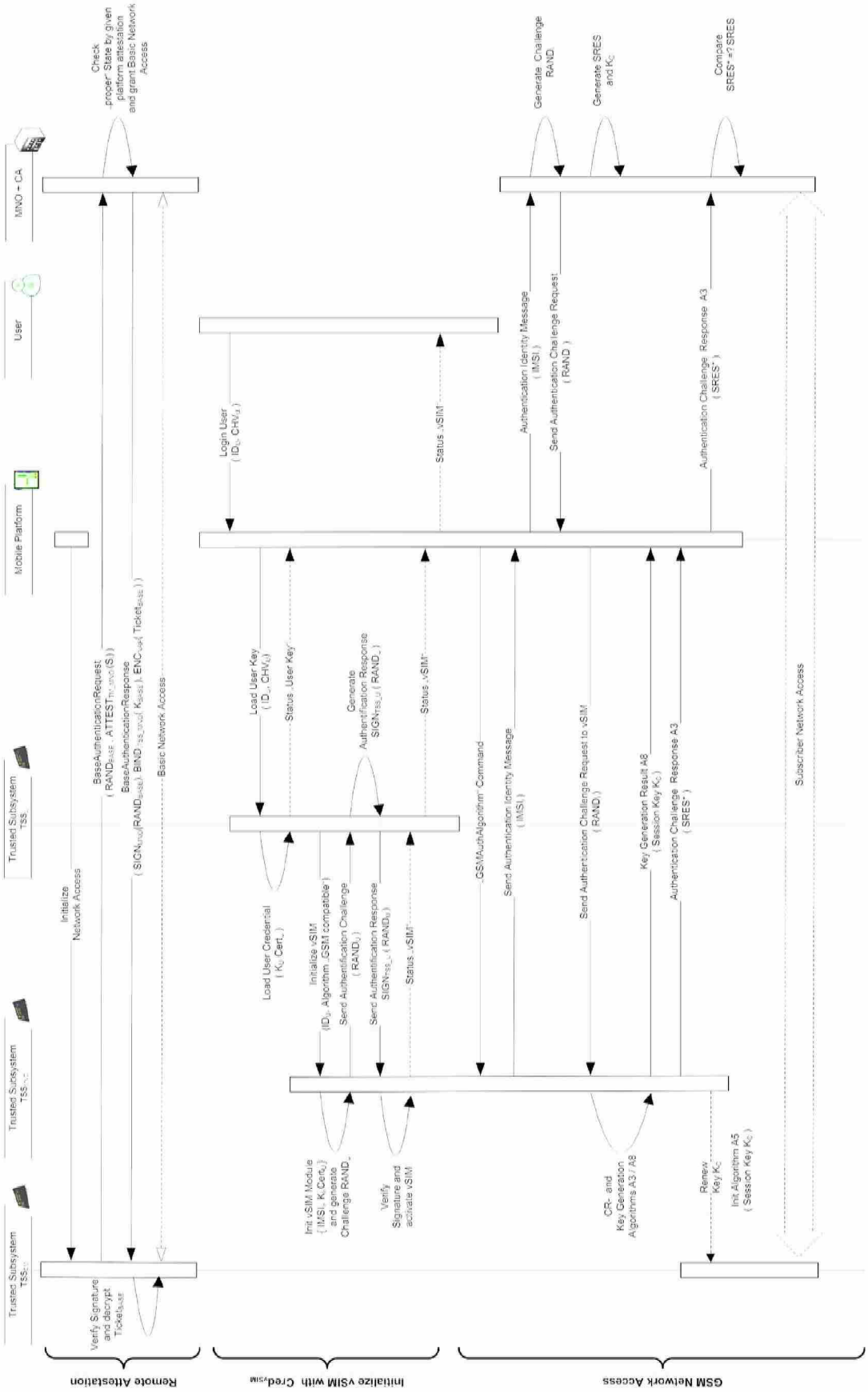

Figure 3.12: *Subscriber Authentication Protocol - Model "Two"*

3.4.3 Model "Three" - Generalized Subscriber Authentication in Network Infrastructures

The following section introduces an authentication model that captures the different aspects of the precedent models from a more abstract point of view. Here, we present a proposal for user- and device authentication using vSIM credentials in generic network infrastructures. This model is based on Trusted Computing and like the predecessor it supports remote attestion and mutual authentication.

In contrast to the previous models, we are using more generalized assumptions and specifications. This concerns particularly the structure and abilities of a vSIM credential $Cred_{vSIM}$ and the trusted $vSIM_{CORE}$ services.

A vSIM credential $Cred_{vSIM}$ is a identity-based identifier that can be used to authenticate a subscriber. It has a unique ID_U of a user U and at least one encryption-based (e.g. symmetric or asymmetric keys) or non-encryption-based (e.g. one-way hash chain) information. Only authorized subjects can create, read or modify to the content of a $Cred_{vSIM}$. A $Cred_{vSIM}$ may hold additional information of a device identity or a set of valid realms

A trusted platform holds a $vSIM_{CORE}$ service, running in an protected environment. $vSIM_{CORE}$ is responsible for the vSIM functionality. In particular, this service implements for the core authentication mechanisms. A specific implementation of mechanisms or protocols depend on the use-case. A $vSIM_{CORE}$ service is able to import (or use external) trusted functionality. Furthermore it holds at least one vSIM credential $Cred_{vSIM}$.

3.4.3.1 Platform and Protocol Precondition

MTP has carried out the boot process and has loaded the specific OS and its trusted services. In particular, this also includes the vSIM services $vSIM_{CORE}$ and $vSIM_{MGMT}$. The MTP has checked, that the installed hardware and running software, are in a trustworthy state and configuration. It is able to report and attest this state, if challenged by an authorized entity.

3.4.3.2 Protocol Scheme and informal Description

Similar to model "Two", the protocol description of this generalized approach is separated into three phases. This generalized setting for subscriber authentication is described by the following sequence of steps.

Phase 1: "Remote Attestation" In this phase, the remote attestation and device authentication are performed as described in Subsection 3.4.2.2. In this general case, the network entities of the MNO are substituted by adequate entities from the generic network infrastructure. Such *adequate entity* may be an authentication server AS within this network.

Phase 2: "Initialization of vSIM Credentials" Initialization of the vSIM services and a vSIM credential are performed as described in Subsection 3.4.1.2. However, this setting is based on the generalized assumption from 3.4.3. Thereby, it provides a wide basis for different types of authentication methods and protocols.

Phase 3: "Subscriber Authentication" The process of subscriber authentication aims to authenticate and authorize a given subscriber to a specified services. In the previous models, the subscriber authentication protocols are based upon a shared secret, which is embedded into the vSIM credential $Cred_{vSIM}$, and a challenge response procedures for authentication. In this generic approach these limitations are in-existent.

Figure 3.13 shows an example of a simple protocol for subscriber authentication with digital signatures based on this generalized model. Here, a random value $RAND_{SRV}$ is used to request an service upgrade at the AS. The TE_{OE} extracts $RAND_{SRV}$ from the $Ticket_{BASE}$, which was obtained in phase 1. Now, the TE_{OE} builds the authentication response $XRES^*_{SRV}$ and signs the $RAND_{SRV}$ with its private signature key $K^{priv}_{TM_{AS}}$. Together with a UID and a service identifier SRV, this signature $XRES^*_{SRV}$ is sent to the AS.

After having received this message, AS verifies signature $XRES^*_{SRV}$. If the signature is valid, the trusted platform is authenticated and a service upgrade will be performed.

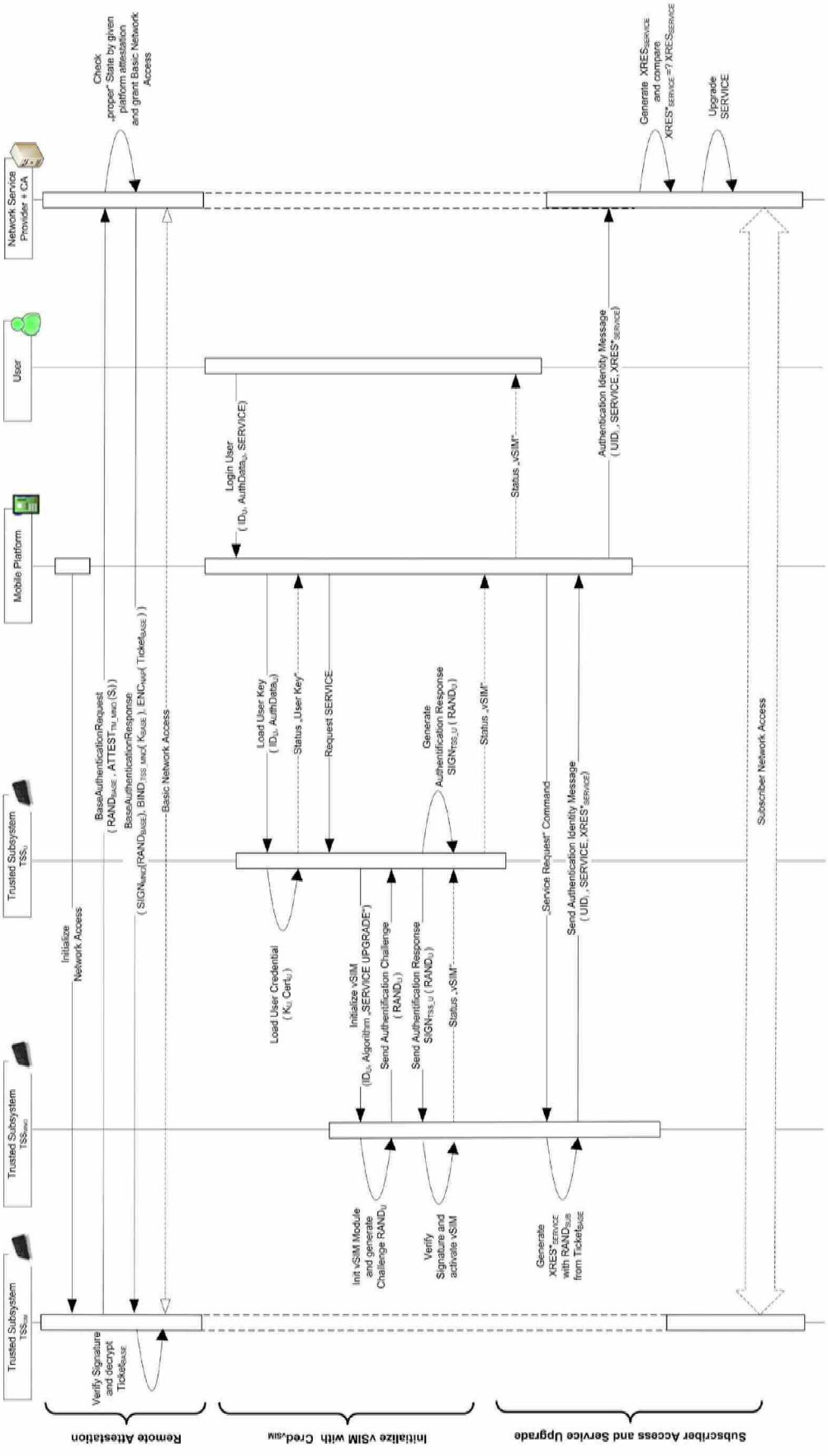

Figure 3.13: *Subscriber Authentication Protocol - Model "Three"*

3.5 Prototypical Implementation of the vSIM Architecture

The prototypical implementation of the trusted engines TE_{DM}, TE_{MNO} and TE_U, and the specified vSIM services will be realized as an extension to an existing trustworthy operating platform. Therefor, we initially introduce to a high-level design of the vSIM architecture in Subsection 3.5.1. Then, in Subsection 3.5.2, we give a short analysis of the significant vSIM platform components.

3.5.1 Overview of the vSIM Platform Design

The high-level design of the proposed vSIM architecture is based on a trustworthy operating platform such as *EMSCB/Turaya* [EMS], or alternatively *IBM sHype* [SVJ+05]. The technical framework consists of four layers:

- *Hardware Layer*,
- *Virtualization Layer*,
- *Trusted Software Layer*, and finally
- *Compartment and Application Layer*.

The platform executes a legacy operating system in coexistence with a running instance of the security architecture. The latter controls a virtual machine with several trusted engines and services compliant to the TCG requirements [Tru06a, Tru06b]. Each trusted engine implements the vSIM services and applications on behalf of a specific stakeholder. The trusted engines are executed within isolated execution environments on top of the security kernel or nested inside TE_{DM}. Figure 3.14 illustrates the platform design of the vSIM architecture.

3.5.2 Analyis of the Platform Components

In the following subsection, we inspect these four platform layers of the vSIM architecture and describe its embedded components.

3.5.2.1 Hardware Layer

The *Hardware Layer* holds the generic hardware components of the computing platform and an additional generic MTM, as described in 2.2.5. The MTM acts as a dedicated master trust-anchor for the complete trusted mobile platform. Although,

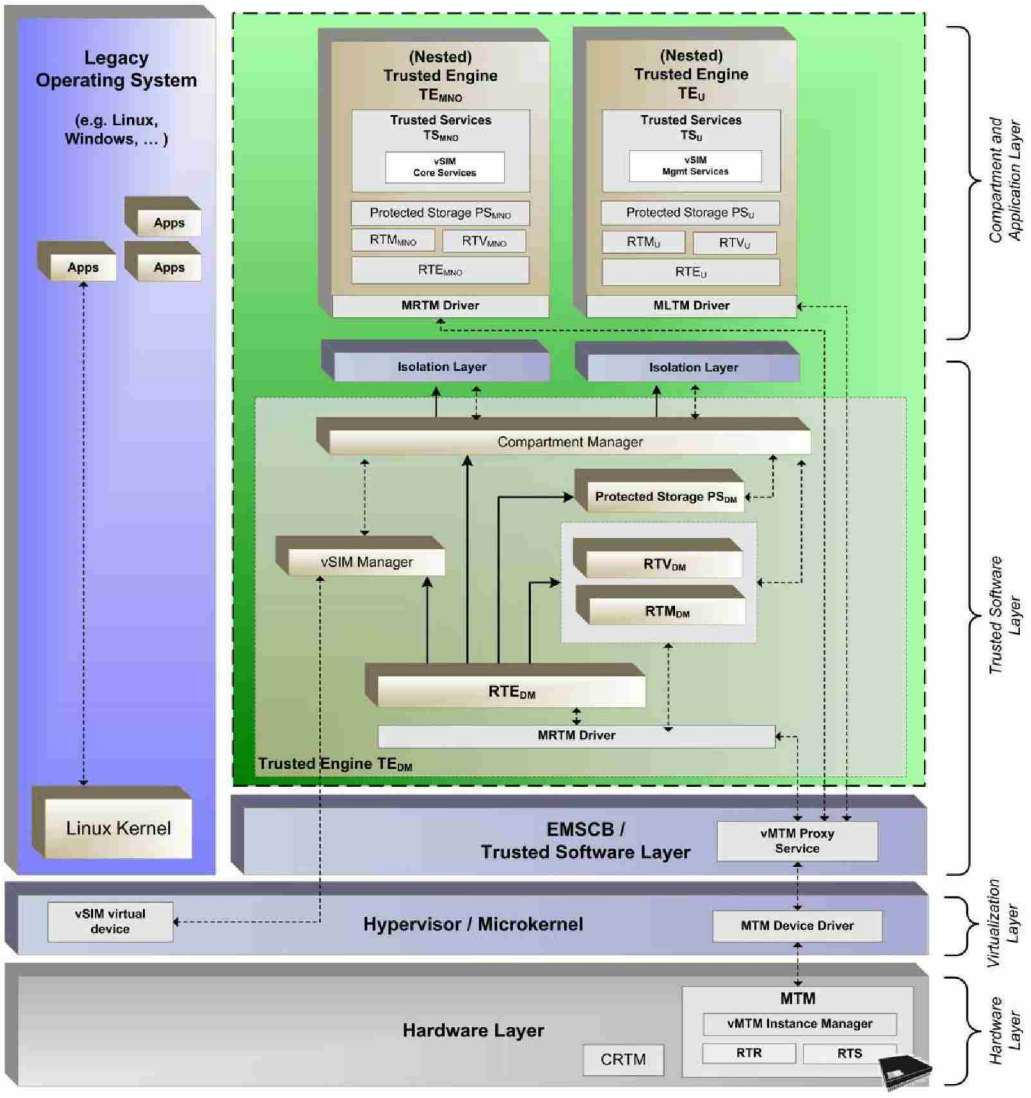

Figure 3.14: *vSIM Architecture on EMSCB/Turaya*

it is favorable to implement this architecture upon a slightly modified MTM, we focus on a (contemporary practicable) solution based on a software-based generic MTM emulation in conjunction with a standard TPM v1.2. The emulator implements a software-based generic MTM emulator based on our proposal in Section 2.2.5. It provides an interface to a standard TPM 1.2 and adapts the hardware-based TPM with MTM functionality compliant to the *TCG MPWG Reference Architecture*.

3.5.2.2 Virtualization Layer

Generic hardware abstraction between the physical hardware and the *Trusted Software Layer* is provided by the *Virtualization Layer*. This layer implements a fully

functional *MTM Host Driver and Device Driver* for a dedicated generic MTM and a *virtual vSIM Device* for communication. Furthermore, it is responsible for instantiation of both, the trusted software layer and the legacy operating system.

- *Virtual vSIM Device* offers an exclusive vSIM communication interface to external entities (e.g. legacy OS). It is implemented by a char-device that provides an external accessible communication channel to the vSIM Manager.

- *MTM Host Driver and Device Driver* implements a full functional software driver that allows interaction with the MTM hardware device and/or the MTM Emulator. It holds the capability to channel the received communication messages to the respective MTM instance.

An excellent fundament is offered by both, the EMSCB project [EMS, PER] and alternatively the sHype project [SVJ+05]. Currently, the EMSCB is based upon a microkernel of the L4-family [BBH+98] that supports instantiation of L4 applications and L4Linux [Hoh96] compartments. On the other hand, sHype is based on the XEN hypervisor [DFH+03], which is able to virtualize a legacy operating system (e.g. Windows, Linux). In general, each solution provides mechanisms for resource management, inter-process-communication, virtual machines, memory management and scheduling.

3.5.2.3 Trusted Software Layer

The *Trusted Software Layer* provides security functionality and is responsible for isolation of embedded applications and software compartments. It implements a set of security services (e.g. trust manager, compartment manager, protected storage manager), which are required by the RTR and RTV, *Protected Storage* and *Compartment Manager* of TE_{DM}. Therefore, it is reasonable to build the significant parts of the device manufacturer engine TE_{DM} within this layer. The components of this layer are:

- *vMTM Proxy Service* is a service that acts as a mediator between the *MTM Device Driver* and a *MTM Client Driver.* When an embedded trusted engine executes a MTM command, the message is routed through this proxy to the requested dedicated *MTM Host Driver.* The proxy will transmit and channel the data to an associated vMTM instance.

- *vMTM Client Driver* implements a restricted software driver that allows interaction with a *vMTM Proxy Service*. It either implements $MRTM$ or $MLTM$ capability. For this reason, a trusted engine is only able to execute its dedicated commands as specified by the TCG MPWG.

- *Roots of Trust (RTE, RTV and RTM* are a set of allocated trusted resources acting on behalf of the stakeholder DM as described in Section 2.2.4.2. In particular, these components are the *Root of Trust for Enforcement*, the *Root of Trust for Verification* and the *Root of Trust for Measurement*.

- *Protected Storage PS* provides storage mechanisms of TE_{DM} as described in Section 2.2.3.4. It acts as a storage manager which uses the protected capabilities and shielded locations of its dedicated vMTM instance $vMTM_{DM}$.

- *Trusted Engines Compartment Manager* controls the instantiation and update of trusted compartments. The main tasks of the compartment manager is to offer a minimal set of required TE_{DM} functionality to the vSIM services, and to route communication messages to their destination. Furthermore, it enforces the isolation of the dedicated trusted engines TE_{MNO} and TE_U that can only be accessed and manipulated by authorized entities.

- *vSIM Manager* represents an interface to the $vSIM_{CORE}$ and $vSIM_{MGMT}$ services and is responsible for the communication with the underlying architecture and external entities.

3.5.2.4 Compartment and Application Layer

The *Compartment and Application Layer* instantiates a set of isolated Trusted Engines TE_σ. Each engine embeds the required allocated trusted resources and protected storage mechanisms as well as the dedicated vSIM services.

- *Trusted Engine* TE_σ are implemented as parallel and isolated Linux ([DFH+03, SVJ+05]) or L4Linux ([EMS, BBH+98, Hoh96]) compartments on behalf of different stakeholders. Each trusted engine is fully equipped with a *MTM Client Driver*, allocated *Roots of Trust* (RTE, RTV and RTM) and *Protected Storage* functionality as described above.

- *VSIM Core Service* $vSIM_{CORE}$ holds the trusted vSIM core services of TSS_{MNO} In particular, it implements the relevant protocol algorithms of MNO. This

includes the subscriber authentication from Section 3.4, and the deployment and management protocols from Section 3.3.

- *VSIM Mgmt Service* $vSIM_{MGMT}$ The $vSIM_{CORE}$ holds the trusted vSIM management services of TSS_U. It implements the relevant protocol algorithms of U. This includes the user portion of the *Subscriber Authentication* protocols from 3.4 and the protocols for *Subscriber Enrollment and vSIM Credential Roll-Out* 3.3.2.

Chapter 4

Benchmark Analysis and Evaluation

The purpose of this *Benchmark Analysis and Evaluation* is to provide a review of the proposed vSIM architecture. A pre-screening of the aspired and existent architectures was conducted to identify the different possibilities for subscriber authentication in mobile cellular networks. We have identified four solutions:

- a single-trust anchor architecture using conventional SIMs,
- a dual trust anchor architecture using conventional SIMs,
- a single trust-anchor architecture using virtual SIMs, and finally
- a client-server architecture using remote SIMs.

It was agreed that the *Single Trust-Anchor Architecture using virtual SIMs* approach turns out as a suitable and sustainable solution, that is able to compete with the SIM-based solutions. This chapter substantiates this assumption.

The evaluation shows that our vSIM architecture can be used to substitute a SIM card. Therefore, we consider the following areas of interest derived from Section 1.2: (1) Security, (2) Cost-effectiveness, (3) Flexibility and Scalability, (4) Portability and Mobility, and (5) Usability, Compatibility and Acceptance.

This chapter is subdivided into three sections. In Section 4.1, we firstly provide a security analysis of the significant protocols of the vSIM architecture. Afterwards, we analyse the further criteria in Section 4.2. Finally, in Section 4.3, a comparison of this architecture with the other solutions is given.

4.1 Security Analysis

An important factor of the security analysis is the protection of the vSIM architecture against attacks. In this context, we have to inspect the following objectives: (1) protection of a MTM, (2) protection of the trusted compartments by the trusted operating system, and (3) the security analysis of the protocols.

4.1.1 Protection Mechanisms of a MTM

Since the MTM is the underlying trust-anchor which provides evidence of the trustworthiness of the vSIM architecture and the associated trusted subsystems TSS_σ, it is required that the MTM itself must be reasonably secured and protected from attacks.

Therefore, the *Common Criteria Protection Profile* makes (still relevant) assertions of the protection requirements of a TPM 1.1 against software and hardware attacks [Tru02a, Tru02b]. This *Protection Profile* stipulates only a limited physical protection of a TPM. However, it is reasonable that the protection against hardware attacks depends on the intended purpose. As a consequence, some TPM/MTM hardware implementations will have stronger physical protections than other [Hew07]. Thus, we assume that a vSIM architecture is built upon a MTM with equal security related characteristics and properties from Section 2.1.3.1, like a conventional SIM card.

4.1.2 Protection Mechanisms of a Trustworthy Operating system

In general, a *Trustworthy Operating System* provides fundamental capabilities that meets the security related requirements from Section 3.2.3 and *TCG MPWG Reference Architecture*. In particular, these operating systems support *Protected Storage*, a tamper-resistant *Isolated Execution Environment*, *Secure Channel* and *Access Control and Authentication*.

4.1.3 Security Analysis of the Protocols

The protocol analysis considers the different security requirements from Section 3.2.3 according to the proposed protocols. In this context, we inspect the provided security mechanism while a vSIM Credential is (1) transfered to the vSIM Container, (2) stored and executed on the mobile trusted platform or (3) transferred between environments of authorized subjects.

4.1.3.1 Protection of the vSIM Credential while on transit

The initial analysis focuses on the confidentiality and integrity protection of the vSIM credential while in transit from the MNO to the destination platform (MTP). This analysis concerns the protocol *Subscriber Enrollment and Credential Roll-Out*, which we have discussed in Section 3.3.2.

An adversary might be able to eavesdrop and modify the protocol messages between MTP and MNO. In order to circumvent resulting attacks, the $Cred_{vSIM}$ is encrypted with a session key K_S. This session key is bound to a specific state of the destination platform using the binding key K_{MNO}^{priv}. An adversary would have to extract K_{MNO}^{priv} from trusted subsystem TSS_{MNO} to recover K_S. Moreover, a HMAC is computed on the digital representation of $Cred_{vSIM}$ using this session key (or a derived/associated integrity key). Therefore, it offers a sufficient degree of confidentiality and integrity, because the decryption is only feasible by an authorized and trusted MTP. If the $Cred_{vSIM}$ was modified by an adversary while transmission to the MTP the computation of the HMAC will fail. However, K_S must also be securely managed and protected by MNO and TSS_{MNO}, at least to the same degree as $Cred_{vSIM}$ itself is protected.

4.1.3.2 Protection of the vSIM Credential while in Storage

The confidentiality and the integrity of the vSIM Credential $Cred_{vSIM}$ while in storage are protected by the mechanisms of protected storage specified in *TCG MPWG Reference Architecture*.

In order to prevent unauthorized access to the vSIM Credential and the associated binding keys, further measures are taken. First, the private portion of the binding key $BK_{TSS_{MNO}}$ is bound to a specific platform configuration of TSS_{MNO} such that this key not loadable until the current environment configuration matches to which the private key was bound. Second, the decryption of $Cred_{vSIM}$ is associated with a successfully challenge-response from $vSIM_{MGMT}$ using a valid signing key SK_{TSS_U}. In conjunction with this, 20 bytes of authorization (see CHV in 2.3) data must be stored with the private signing key SK_U^{priv}. This meets the *Strong Authentication* and *Portability and Mobility* requirements from 2.1.3.1.

4.1.3.3 Protection of the vSIM Credential during execution

The confidentiality and the integrity of the vSIM Credential $Cred_{vSIM}$ while in storage are protected by the mechanisms of protected execution environment. The vSIM services which are running within that environment of TSS_{MNO} and TSS_U can not be read or manipulated by an unauthoritzed entity.

Nevertheless, the vSIM Credential is only protected by software mechanisms after the object is loaded successfully into the execution environment. In terms of security, a dedicated hardware-based protection of security-sensitive data is stronger than the software-based solution. And consequently, if the protection of the protected execution environment fails, the vSIM Credential may be accessible to an adversary. In order to circumvent this security flaw, a dedicated MTM could also be equipped with A3/A8 GSM computation engine, as shown in Figure 4.1. Consequently, the secret individual key K_i would never leave the hardware protected environment of the MTM.

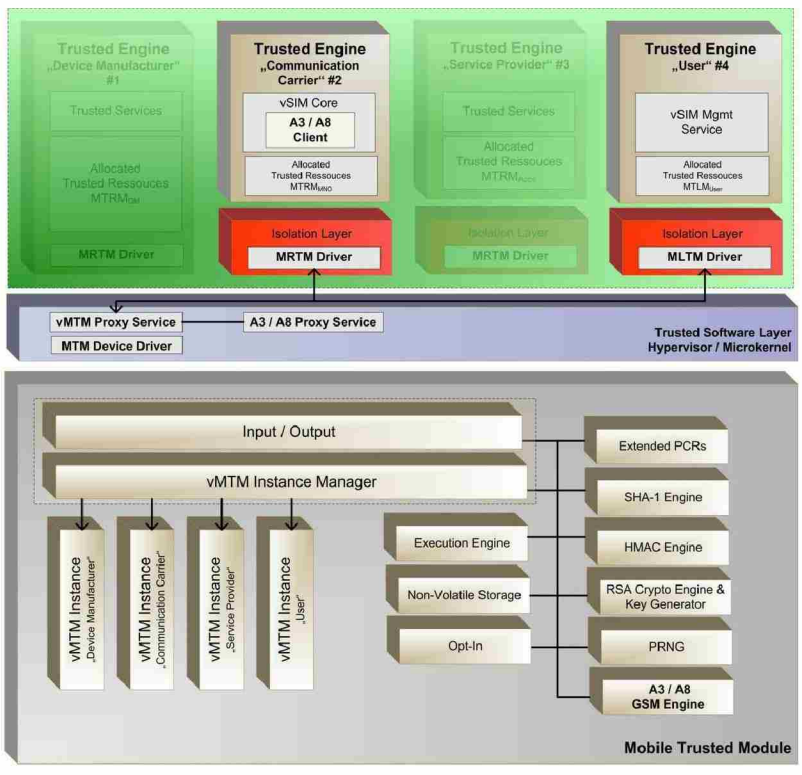

Figure 4.1: *Generic MTM architecture with an additional A38 engine*

A $vSIM_{CORE}$ service sends its GSM authentication request to a *A3/A8 Proxy Service*, which is a part of the *vMTM Proxy Service* within the *Trusted Software*

Layer. This service forwards the messages from the $vSIM_{CORE}$ to the dedicated MTM instance.

4.2 Benchmark Analysis

The analysis of the other benchmark criteria and conditions of Section 1.2 is another important issue for evaluation of the vSIM architecture. For this reason, we have to inspect the following objects: (1) Cost-effectiveness, (2) Flexibility and Scalability, (3) Portability and Mobility, (4) Usability, (5) Compatibility and (6) Acceptance.

4.2.1 Cost-effectiveness

The major aspect of our solution concerns cost effectiveness. Our approach reduces the production and logistic costs, regarding to the two-trust-anchor solution while keeping an adequate level of security and trustworthiness. Unfortunately, it is difficult to determine an exact and sustainable cost analysis, since these costs are treated strictly confidential by device manufacturers.

- *Manufacturing and Production Costs:* The main component that affects the costs of a security chip is naturally the micro controller. With the proposed solution, the MNO is able to use the already installed MTM for its purpose of subscriber authentication and protected storage. The mobile device is not necessarily equipped with a conventional SIM card.

- *Logistic Costs:* As the vSIM Credentials are completely implemented as a software object, it implies that a $Cred_{vSIM}$ can transferred from the MNO to the vSIM Container via an arbitrary network connection. For this reason, a MNO can reduce and minimize effective costs of the logistic process.

4.2.2 Flexibility and Scalability

The vSIM service architecture for subscriber authentication offers flexibility and scalability to both, the user and the MNO. The next paragraphs shortly describes the benefits concerning this issue.

Parallel vSIMs in a single Mobile Device The proposed vSIM architecture supports the instantiation and usage of many parallel vSIM services. It conveniently allows a subscriber to make and receive calls on different $MSISDN$ numbers with

different subscriber identities, from one single handset without switching a physical SIM card.

Online Registration and Roll-Out of vSIM Credentials A device owner initiates and controls the subscriber registration, vSIM credential roll-out and installation process as described in Subsection 3.3.2. This is potentially done by using a web-based interface giving subscribers full control to decide how they would like to register themself and receive their vSIM Credential on their mobile devices.

Network-based Migration of a vSIM Container and vSIM Credential The vSIM Container is migrateable between different devices holding the same stakeholder and a suitable platform configuration as described in Section 3.3.3. A device owner/user can migrate a vSIM Container and the vSIM Credential using an arbitrary network connection. This implies migration over an existing PAN (e.g. Bluetooth) as well as an established connection to the destination device over the internet.

Remote Update of Services, Firmware and Applications Since the MNO is able to access its dedicated compartment TSS_{MNO}, arbitrary data object can be installed or modified by an authorized entity. Thus data object, for instance are either vSIM service- and algorithm updates, or new firmware and applications for the mobile device.

Dynamic Up- and Downgrade of Network Services The vSIM architecture offers the potential for dynamic up-/downgrade of network services and finer-grained functional restriction. These functionalities provide mobile communication services as well as location-based services, like content services or mobile payment services. Since some of these services require a higher level on platfrom integrity and conformity (e.g. online-banking transaction) the vSIM architecture is able to provide evidence of the platform state. In particular, the approach for subscriber authentication in Model "Two" (see 3.4.2) allows to add or remove functionality and services easily.

4.2.3 Portability and Mobility

The vSIM architecture allows subscribers to use a $Cred_{vSIM}$ with arbitrary trusted mobile platform. With the identified protocols for deployment and management

from Section 3.3, a vSIM credential is removeable and portable to other devices. Hence, it enables a subscriber to use its credential with other devices, and vice versa.

Using two different vSIM compartments on behalf of the stakeholder MNO and U, the vSIM architecture enables explicitly to differentiate between device and subscriber identity. A MTP and a subscriber have their own identity with different intended usage characteristics.

4.2.4 Usability, Compatibility and Acceptance

Determining usability and compatibility of a system is an important part, since it finally leads to the acceptance of the proposed vSIM architecture. For this reason, we have designed the subscriber authentication protocols as well as the protocols for deployment and management with a high level of compatibility and usage characteristics to current GSM standard. This was reached by using algorithms, commands and grammar compliant to GSM 11.11 and GSM 11.14 standard from Section 2.1.3. The proposed model "One" for subscriber authentication, in particular is compliant to this GSM standard.

4.3 Comparison and Evaluation

A comparison of the different architecture as well as the proposed models for subscriber authentication from Section 3.4 has been done using the set of benchmarks from Section 1.2. Table 4.2 illustrates the results.

In the present thesis, we have substantiated our choice and have proven that a sufficient level of each criteria is reached. In this context, it is reasonable to decide in favor of the proposed *Single Trust-Anchor Architecture using virtual SIMs for Subscriber Authentication* in conjunction with "Model One" based on a generic MTM emulation. The tradeoff and fundamental design decisions are primarily based on the cost-value ratio by reaching an adequate level of security and better deployment and management functionality in comparison with the other competing solutions.

Although, the proposed Models "Two" and "Three" offer more flexibility and security by embedding platform attestation. The decision is resulted by the compatibility to current GSM standard. Thus, a higher level of acceptance from both, the mobile network operator and the device manufacturer is reachable.

Architecture / Criteria	SIM	SIM / MTM	vSIM Architecture: Model "One"	vSIM Architecture: Model "Two"	vSIM Architecture: Model "Three"	C/S SIM
Security						
Protected Storage	X	X	X	X	X	X
Isolated Execution Environment	X	X	X	X	X	X
Secure Channel	X	X	X	X	X	X
Access Control and Authentication	X	X	X	X	X	X
Remote- / Platform Attestation				X	X	X
Trustworthy Operating Client Platform		X	X	X	X	
Resistant against Software Attacks	X	X	X	X	X	X
Resistant against Hardware Attacks	X	X	X	X	X	X
Mutual Authentication	X	X		X	X	
Hardware Protected K_i	X	X	-/X	X	-/X	X
Cost Effectiveness						
Reduction of Manufacturing Costs			X	X	X	
Reduction of Logistic Costs			X	X	X	
Portability and Mobility						
Device Mobility	X		X	X		X
User Mobility	X		X	X		X
Diversity of User and Device Identity	X		X	X	X	X

Architecture / Criteria	SIM	SIM / MTM	vSIM Architecture: Model "One"	vSIM Architecture: Model "Two"	vSIM Architecture: Model "Three"	C/S SIM
Flexibility and Scalability						
Multiple SIMs in a single Device			X	X	X	X
Remote Service Update	-/X	-/X	X	X	X	
Dynamic Service Up- and Downgrade				X	X	
Remote Take-Ownership vSIM Container			X	X	X	
Network-based Migration			X	X	X	X
Usability						
Conventional Usage Characteristics	X	X	X	X	X	
Online-Registration of a Subscriber	X	X	X	X	X	
Online-Roll-Out of SIM Credentials			X	X	X	
Compatibility						
Compatibility to GSM 01.02	X	X	X			X
Compatibility to GSM 11.11	X	X	X	X		X
Compatibility to GSM 11.14	X	X	X	X		X
Acceptance						
Device Owner / User	X	X	X	X		
Mobile Network Operator (MNO)	X	X	X	-/X		

Table 4.2: *Comparison vSIM/SIM Architectures*

Chapter 5

Conclusions and further work

In this thesis, we have examined both, theoretical and practical aspects of how subscriber authentication in mobile cellular networks could be implemented to the next generation of mobile phones and devices. This chapter summarizes the results and discusses possible further work.

5.1 Conclusion

The present thesis demonstrates the substitutability of a SIM card with an adequate trusted software module, supported and protected by a trustworthy operating system. In this regard, we have introduced vSIM Credentials as a means for subscriber authentication based on the TCG MPWG technology. It offers a real alternative to the other SIM-based solutions under consideration, while an sufficient degree of security and usage characteristics are reached.

The first contribution of this thesis has examinated several architectural directions, including our aspired architecture of subscriber authentication with vSIMs. Furthermore, we have discussed a set of benchmarks, which have been seen crucial in terms of our objective.

The required theoretical fundament and background of GSM and Trusted Computing was given in Chapter 2. In particular, we have discussed the forthcoming *TCG MPWG Reference Architecture* and have detailed theoretical and practical aspects of this specification.

In Chapter 3, we have identified and developed a comprehensive framework for subscriber authentication in Chapter 3. This framework includes the fundamental vSIM architecture including essential methods for deployment and management as well as the conceptual models for subscriber authentication in mobile cellular net-

works. Therefor, we have systematically discussed the procedures and mechanisms of efficient administration, management and maintenance of subscriber credentials. The following protocols were detailed:

- Remote-Take-Ownership of a vSIM Container
- Subscriber Enrollment and vSIM Credential Roll-Off, and
- Migration of a vSIM Container and vSIM Credential

Based on this fundament, we have developed three intergraded models for subscriber authentication using trusted computing.

- Model "One": Subscriber Access in mobile cellular Networks based on Trusted Computing with compatibility to GSM - Authentication
- Model "Two": GSM-Subscriber Authentication in mobile cellular Networks based on Trusted Computing with Remote Attestation for Restricted-Network-Access
- Model "Three": Generalized Subscriber Authentication in IT Networks Infrastructures based Trusted Computing with Remote Attestation for Restricted-Network-Access

This vSIM architecture was analysed and evaluated in Chapter 4. As a result, it shows that a traditional SIM-Card could be replaced by a virtual SIM which is base on this framework. In particular, Table 4.2 reveals that the proposed *Single Trust-Anchor Architecture using virtual SIMs for Subscriber Authentication* in conjunction with "Model One" based on a generic MTM emulation is a suitable and sustainable approach with regard to the SIM-based solutions. A prototypical implementation on a trustworthy operating platform is under development.

5.2 Outlook and further Research

From a general point of view it seems like Trusted Computing will play a significant role in future computing. Using a vSIM as a trusted and protected software allows expansion to a much wider field of authentication and identification management systems on standard PC platforms [DEPY03]. The realization of (mobile) trust credentials in user-centric scenarios by vSIM credentials, as shown in Figure 5.1, or the support of online transactions by vSIM authentication are thinkable approaches.

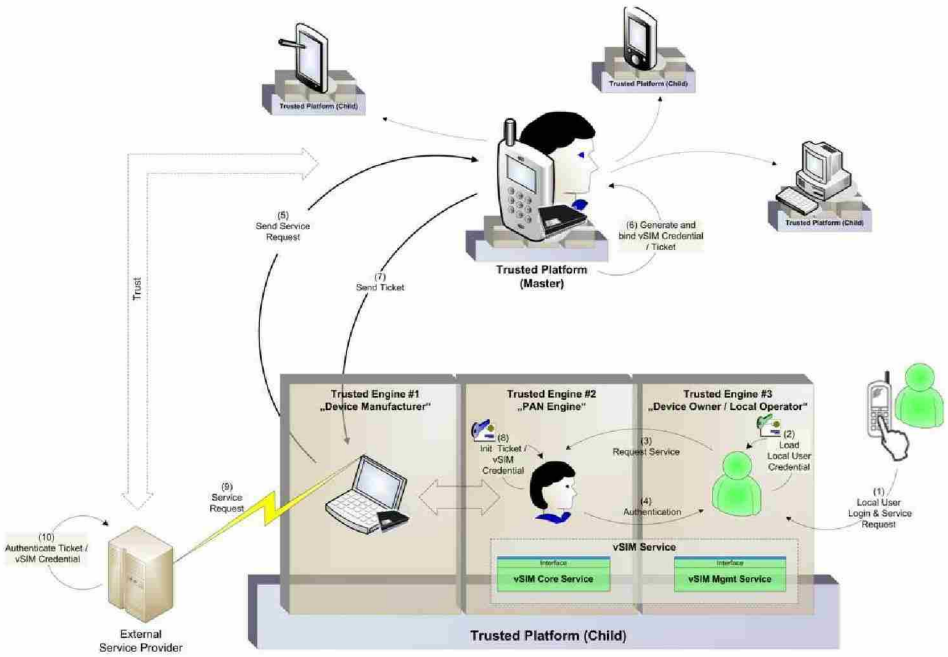

Figure 5.1: *Personal-Area-Networks and vSIMs*

For this reason, we plan to integrate the vSIM model into the generic domain. However, there are some privacy and security challenges associated with this implementation on a desktop computer using an unmodified TPM, which needs a further research.

Bibliography

[BBH+98] R. Baumgartl, M. Borriss, Cl.-J. Hamann, M. Hohmuth, L. Reuther, S. Schönberg, and J. Wolter. *Dresden Realtime Operating System (DROPS)*. In *Workshop of System-Designed Automation*, 1998. (SDA'98).

[BCG+06] S. Berger, R. Caceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn. vTPM: Virtualizing the Trusted Platform Module. Technical report, IBM T. J. Watson Research Center, Yorktown Heights, 2006.

[DEPY03] J. Dashevsky, E. C. Epp, J. Puthenkulam, and M. Yelamanchi. SIM Trust Parameters. *Intel Developer Update Magazine*, 2003.

[DFH+03] B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, I. Pratt, A. Warfield, P. Barham, and R. Neugebauer. *Xen and the Art of Virtualization*. In *Proceedings of the ACM Symposium on Operating Systems Principles*, 2003.

[Eck04] C. Eckert. *IT-Sicherheit - Konzepte, Verfahren und Protokolle*, volume 2, chapter 13.1 (GSM), pages 641–643. Oldenbourg Verlag, Munich, 2004.

[Eck06] C. Eckert. *IT-Sicherheit - Konzepte, Verfahren und Protokolle*, volume 4, chapter 11.10 (Trusted Computing), pages 617–649. Oldenbourg Verlag, Munich, 2006.

[EMS] EMSCB - Towards Trustworthy Systems with Open Standards and Trusted Computing. Offical Website. http://www.emscb.com.

[Hew07] Hewlett-Packard Development Company, L.P. Business PC Security Solutions - Questions and Answers. http://h20331.www2.hp.com/Hpsub/cache/292232-0-0-225-121.html, 2007.

[Hoh96] M. Hohmuth. *Linux-Emulation auf einem Mikrokern.* PhD thesis, Technical University Dresden, Dresden, 1996.

[iGR06] iGR. *Worldwide Wireless and Mobile Market Forecast, 2005-2010.* Technical report, iGillott Research Inc., 2006.

[Imp] Implementa SIM Server. Offical Website. http://www.implementa.com.

[JK03] J. Jonsson and B. Kaliski. *Public-Key Cryptography Standards PKCS-1.* Technical Report 2.1, RSA Laboratories, Bedford, 2003.

[KBC97] H. Krawczyk, M. Bellare, and R. Canetti. *HMAC: Keyed-Hashing for Message Authentication.* Technical Report RFC-2104, Internet Engineering Task Force (IETF), 1997.

[KS06] N. Kuntze and A. U. Schmidt. *Trusted Computing in Mobile Action.* In *Peer reviewed Proceedings of the ISSA 2006 From Insight to Foresight Conference.* Information Security South Africa (ISSA), 2006.

[Mit05] C. Mitchell. *Trusted Computing.* The IEE, 2005. IEE Professional Applications of Computing Series 6.

[Nat02] National Institute of Standards and Technology. *FIPS PUB 180-2; Specifications for the Secure Hash Standard.* Technical report, NIST, 2002. Federal Information Processing Standards (FIPS).

[NYHR05] C. Neuman, T. Yu, S. Hartman, and K. Raeburn. *The Kerberos Network Authentication Service (V5).* Technical Report RFC-4120, Internet Engineering Task Force (IETF), 2005.

[PER] Perseus Security Framework. Offical Website. http://www.perseus-os.org.

[rGPP91] 3rd Generation Partnership Project. *3GPP TS 03.20; Security-related Network Functions.* Technical Report 3.3.2, 3GPP, 1991. Technical Specification Group Services and System Aspects.

[rGPP97a] 3rd Generation Partnership Project. *3GPP TS 02.09; Security Aspects.* Technical Report 6.1.0, 3GPP, 1997. Technical Specification Group Services and System Aspects.

[rGPP97b] 3rd Generation Partnership Project. *3GPP TS 11.11; Specification of the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface.* Technical Report 3.3.2, 3GPP, 1997. Technical Specification Group Services and System Aspects.

[rGPP01] 3rd Generation Partnership Project. *3GPP TS 01.02; General description of a GSM Public Land Mobile Network (PLMN).* Technical Report 6.1.0, 3GPP, 2001. Technical Specification Group Services and System Aspects.

[rGPP02] 3rd Generation Partnership Project. *3GPP TS 55.205; Specification of the GSM-MILLENAGE Algorithms; An example algorithm set for the GSAuthentication and Key Generation functions A3 and A8.* Technical Report 6.0.0, 3GPP, 2002. Technical Specification Group Services and System Aspects.

[rGPP04] 3rd Generation Partnership Project. *3GPP TS 11.14; Specification of the SIM Application Toolkit (SAT) for the Subscriber Identity Module - Mobile Equipment (SIM-ME) interface.* Technical Report 8.17.0, 3GPP, 2004. Technical Specification Group Services and System Aspects.

[rGPP06] 3rd Generation Partnership Project. *3GPP TS 55.216; Specification of the A5/3 Encryption Algorithm for GSM, and the GEA Encryption Algorithm for GPRS.* Technical Report 6.1.0, 3GPP, 2006. Technical Specification Group Services and System Aspects.

[rGPP07] 3rd Generation Partnership Project. *3GPP TS 21.101: Technical Specifications and Technical Reports for a UTRAN-based 3GPP system.* Technical Report 6.6.0, 3GPP, 2007. Technical Specification Group Services and System Aspects.

[Sma01] Smart Card Securtiy User Group. *SCSUG-SCPP; Smart Card Protection Profile.* Technical report, Common Criteria for Information Technology Security Evaluation, 2001.

[Str05] M. Strasser. *Software-based TPM Emulator for Linux.* http://tpm-emulator.berlios.de/, 2005.

[SVJ+05] R. Sailer, E. Valdez, T. Jaeger, R. Perez, L. van Doorn, J. L. Griffin, and S. Berger. *sHype: Secure Hypervisor Approach to Trusted Virtualized Systems.* Technical Report RC23511, IBM Research Division, 2005.

[TCG] TCG. Offical Website. https://www.trustedcomputinggroup.org.

[Tru02a] Trusted Computing Group. *Trusted Computing Platform Alliance (TCPA) Main Specification.* Technical Report 1.1b, Trusted Computing Group, 2002.

[Tru02b] Trusted Computing Group. *Trusted Platform Module Protection Profile.* Technical Report 1.9.7, Trusted Computing Platform Alliance (TCPA), 2002.

[Tru05] Trusted Computing Group. *TPM Main Part 1 Design Principles.* Technical Report Version 1.2 Revision 94, TCG, 2005.

[Tru06a] Trusted Computing Group. *TCG MPWG Mobile Reference Architecture.* Version 1.0 Draft 28, currently unpublished (30.04.2007), 2006.

[Tru06b] Trusted Computing Group. *TCG MPWG Mobile Trusted Module Specification.* Technical Report Version 0.9 Revision 1, TCG, 2006.

[Tru06c] Trusted Computing Group. *TCG Software Stack (TSS).* Technical Report Version 1.2, Level 1, Errata A, TCG, 2006.

[Tru07] Trusted Computing Group. *TCG Specification Architecture Overview.* Technical Report Revision 1.3, TCG, 2007.

[Wal00a] B. Walke. *Mobilfunknetze und ihre Protokolle. Grundlagen, GMS, UMTS und andere zellulare Mobilfunknetze*, volume 2, chapter 5, pages 367–458. B.G. Teubner, Stuttgart, 2000.

[Wal00b] B. Walke. *Mobilfunknetze und ihre Protokolle. Grundlagen, GMS, UMTS und andere zellulare Mobilfunknetze*, volume 2, chapter 3, pages 135–342. B.G. Teubner, Stuttgart, 2000.

[Wir06] Wireless Inteligence. *GSM subscriber statistics - GSMA Q2-2006.* Technical report, GSMA, 2006.

Erklärung

Hiermit versichere ich, die vorliegende Diplomarbeit selbstständig und unter ausschliesslicher Verwendung der angegebenen Quellen und Hilfsmittel angefertigt zu haben. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mannheim, 30.04.2007 Michael Kasper